#include<iostream> using namespace std;

int x[101],a[101]; // x cate bancnote de un fel, a valoarile bancnotelor

int n,s,Suma,nr;

void tipar(int k) { int i; nr++;

cout<<"Solutia "<<nr<<":"<<endl;

for(i=1;i<=k-1;i++) { if(x[i]>0) cout<<x[i]<<" \* "<<a[i]<<" + "; }

if(x[k]>0) cout<<x[k]<<" \* "<<a[k]; cout<<endl; }

void back(int k) {

if(s==Suma)tipar(k-1);

else { x[k]=-1;

while(x[k]\*a[k]+s<Suma && k<n+1) { x[k]++;

**s=s+x[k]\*a[k];**

back(k+1);

**s=s-x[k]\*a[k];**  } } }

int main() { int i; cout<<"Suma="; cin>>Suma; cout<<"nr bancnote ="; cin>>n;

for(i=1;i<=n;i++) { cout<<"a["<<i<<"]="; cin>>a[i]; }

back(1); return 0; }

/\* Plata unei sume cu bancnote de valori date. Se dau o suma şi n tipuri de bancnote având valori de a1,a2,...,an lei. Se cer toate modalităţile de plată a sumei utilizând aceste bancnote. Presupunem că dispunem de un număr suficient de bancnote de fiecare tip.

Exemplu: Pentru Suma=5 , n=3 (trei tipuri de bancnote) cu valorile 1,2,3, soluţiile sunt:

Sol 1: 1 de 2, 1 de 3 Sol 2: 1 de 1, 2 de 2 Sol 3: 2 de 1, 1 de 3 Sol 4: 3 de 1, 1 de 2 Sol 5: 5 de 1

Reprezentarea soluţiei: - soluţia are n componente

-fiecare componentă xk are valori în mulţimea {0,1,2,….,Suma/ak}

Condiţia de validare: - suma parţială s=a[1]\*x[1]+a[2]\*x[2]+…+a[k]\*x[k] < Suma

-suma parţială s se actualizează la adăugarea sau eliminarea unei valori x[k] Varianta recursivă\*/